

Application No. : 09/242,383
Amdt. Dated : January 15, 2004
Reply To Office Action Of : August 15, 2003

Amendments To The Claims

The listing of claims will replace all prior versions and listings of claims in the application. The listing of claims present each claim with its respective status shown in parentheses.

Listing of Claims

Claims 1-38 (canceled)

Claim 39 (new): An object oriented operating system programmed using an object oriented programming language which defines a first class, the object oriented operating system comprising a second class defined by the operating system to replace the first class defined by the programming language, wherein the second class utilizes different memory management functionality than the first class.

13
B
Claim 40 (new): An object oriented operating system programmed using an object oriented programming language which defines a class, the object oriented operating system comprising a plurality of classes defined by the operating system to replace the class defined by the object oriented programming language, wherein at least one of the plurality of operating system defined classes utilize different memory management functionality than the class defined by the object oriented programming language. 112/21

Claim 41(new): The object oriented operating system of Claim 40, wherein the plurality of classes comprises a first class, a second class, and a third class.

Claim 42 (new): The object oriented operating system of Claim 41, wherein the first class is configured to reduce memory usage by storing an address for data rather than the data itself.

Claim 43 (new): The object oriented operating system of Claim 41, wherein the second class is configured to handle data that is size limited.

Application No. : 09/242,383
Amdt. Dated : January 15, 2004
Reply To Office Action Of : August 15, 2003

Claim 44 (new): The object oriented operating system of Claim 41, wherein the third class is configured to provide sophisticated memory management tasks.

Claim 45 (new): The object oriented operating system of Claim 40, wherein the plurality of classes are implemented as flat structures.

Claim 46 (new): The object oriented operating system of Claim 40, wherein the plurality of classes are implemented as polymorphic objects.

Claim 47 (new): An object oriented operating system programmed using an object oriented programming language which defines at least one class, the object oriented operating system comprising a set of three classes defined by the operating system to replace the class defined by the object oriented programming language, wherein each of the three classes utilize different memory management functionality from the class defined by the object oriented programming language, and wherein the first class is configured to reduce memory usage by storing an address for data rather than the data itself, the second class is configured to handle data that is size limited, and the third class is configured to provide sophisticated memory management tasks .

Claim 48 (new): An object oriented operating system programmed using an object oriented programming language that includes a string class, wherein the string class is redefined as a set of string descriptor classes, and wherein at least one of the description classes within the set of string descriptor classes utilizes different memory management functionality.

Claim 49 (new): The object oriented operating system of Claim 48, wherein the set of string descriptor classes comprises a first string descriptor class, a second string descriptor class, and a third string descriptor class.

Claim 50 (new): The object oriented operating system of Claim 49, wherein the first string descriptor class is configured to use less memory than the string class.

Claim 51 (new): The object oriented operating system of Claim 49, wherein the first string descriptor class is configured to reduce memory usage by storing an address for data rather than the data itself.

Claim 52 (new): The object oriented operating system of Claim 50, wherein the data is literal text stored in an original memory location.

Application No. : 09/242,383
Amdt. Dated : January 15, 2004
Reply To Office Action Of : August 15, 2003

Claim 53 (new): The object oriented operating system of Claim 49, wherein the second string descriptor class is configured to be size limited.

Claim 54 (new): The object oriented operating system of Claim 53, wherein the second string descriptor class is configured to provide a limited sub-set of memory management functions available in the string class.

Claim 55 (new): The object oriented operating system of Claim 49, wherein the second string descriptor class is configured to use static memory.

Claim 56 (new): The object oriented operating system of Claim 49, wherein the second string descriptor class is configured to store the actual referenced data.

Claim 57 (new): The object oriented operating system of Claim 49, wherein the third string descriptor class is configured to provide sophisticated memory management tasks.

Claim 58 (new): The object oriented operating system of Claim 57, wherein the third string descriptor class is further configured to support fully dynamic text.

Claim 59 (new): The object oriented operating system of Claim 57, wherein the third string descriptor class is configured to use heap memory.

Claim 60 (new): The object oriented operating system of Claim 48, wherein the set of string descriptor classes are implemented as flat structures.

Claim 61 (new): The object oriented operating system of Claim 48, wherein the set of string descriptor classes are implemented as polymorphic objects.

Claim 62 (new): The object oriented operating system of Claim 48, wherein the set of string descriptor classes comprise a function that is responsive to a value in a predetermined field within the polymorphic objects.

Claim 63 (new): The object oriented operating system of Claim 62, wherein the set of string descriptor classes are configured such that the field shares a machine word with another data item.

Claim 64 (new): The object oriented operating system of Claim 48, wherein the set of string descriptor classes include a statement defining the length of data referenced.

Claim 65 (new): The object oriented operating system of Claim 48, wherein the object oriented operating system is written in an 8 bit character set and a 16 bit character set

Application No. : 09/242,383
Amdt. Dated : January 15, 2004
Reply To Office Action Of : August 15, 2003

invariant form by using aliases for class names that are 8 bit character set and 16 bit character set invariant.

Claim 66 (new): The object oriented operating system of Claim 65, wherein the object oriented operating system is written in code invariant form by using aliases for class names that are code invariant.

Claim 67 (new): The object oriented operating system of Claim 48, wherein the set of string descriptor classes are inherently length specified and hence have no '0' terminator.

Claim 68 (new): A computing device programmed with an object oriented operating system, wherein the object oriented operating system is programmed using an object oriented programming language that includes a string class, wherein the string class is redefined as a set of string descriptor classes, and wherein at least one of the string descriptor classes of the set of string descriptor classes utilizes different memory management functionality.

Claim 69 (new): A peripheral device for a computer programmed with an object oriented operating system, wherein the object oriented operating system is programmed using an object oriented programming language that includes a string class, wherein the string class is redefined as a set of string descriptor classes, wherein at least one of the string descriptor classes of the set of string descriptor classes utilizes different memory management functionality, and wherein peripheral device is programmed to handle objects that fall into the set of string descriptor classes.

Claim 70 (new): A computer readable media encoded with an object oriented operating system, wherein the object oriented operating system is programmed using an object oriented programming language that includes a string class, wherein the string class is redefined as a set of string descriptor classes, and wherein at least one of the string descriptor classes of the set of string descriptor classes utilizes different memory management functionality.

Claim 71 (new): A method of programming an object oriented operating system in an object oriented programming language that includes a first class, the method comprising:
redefining the first class as a set of classes; and
utilizing a different memory management functionality for at least one class in the set of classes.

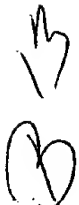
Claim 72 (new): The method of Claim 71, wherein the first class is a string class.

Application No. : 09/242,383
Amdt. Dated : January 15, 2004
Reply To Office Action Of : August 15, 2003

Claim 73 (new): A method of operating a micro-processor, the method comprising:
programming an object oriented operating system in an object oriented programming language that includes a first class;
redefining the first class as a set of classes;
utilizing a different memory management functionality for at least one class in the set of classes; and
executing the object oriented operating system on the micro-processor.

Claim 74 (new): The method of Claim 73, wherein the first class is a string class.

Claim 75 (new): A computing device programmed to manipulate or access objects of the string class using an object oriented operating system, wherein the objects of the string class are derived from a single base class and the operating system handles all such objects of the string class according to one or more of the following requirements:

- 
- (a) objects of the string class for literal text are handled as belonging to a class in which a pointer points to the memory location where the text string is stored;
 - (b) objects of the string class for length limited text are handled as belonging to a class in which a buffer stores text of a predetermined length requiring a limited subset of available memory management functions; and
 - (c) objects of the string class using heap memory are handled as belonging to a class requiring the full set of available memory management functions.

Claim 76 (new): The computing device of Claim 75, further comprising a program which interfaces with the operating system and which also handles objects according to one or more of the requirements.

Claim 77 (new): The computing device of Claim 75, wherein objects satisfying one or more of the requirements are flat structures.

Claim 78 (new): The computing device of Claim 75, wherein objects of the string class for length limited text are stored in particular memory locations at run time which are not part of the heap memory.

Claim 79 (new): The computing device of Claim 75, wherein the objects are polymorphic.

Application No. : 09/242,383
Amdt. Dated : January 15, 2004
Reply To Office Action Of : August 15, 2003

Claim 80 (new): The device of Claim 79, wherein polymorphism is achieved by providing a data field for each object which identifies its class, with a different action being associated with different data field values.

Claim 81 (new): The computing device of Claim 80, wherein the data field is a part of the representation of another data item within a machine word.

Claim 82 (new): The computing device of Claim 81, wherein the same source code is used, irrespective of the character code system and character code width being used, by using aliases for class names that are character code independent.

Claim 83 (new): The computing device of Claim 82, wherein the source code using text strings is written in a manner independent of the strings' actual ASCII or Unicode implementation by using a system of aliases for class names.

Claim 84 (new): The computing device of Claim 75, wherein objects have information on the length of the data they contain and hence have no '0' terminator.

Claim 85 (new): A method of allowing objects of the string class to be manipulated or accessed by a program using an object oriented operating system, wherein the program handles all such objects according to one or more of the following requirements:

- (a) objects of the string class for literal text are handled as belonging to a class in which a pointer points to the memory location where the text string is stored;
- (b) objects of the string class for length limited text are handled as belonging to a class in which a buffer stores text of a predetermined length requiring a limited subset of available memory management functions; and
- (c) objects of the string class using heap memory are handled as belonging to a class requiring the full set of available memory management functions.

Claim 86 (new): The method of Claim 85, being performed by an operating system.

Claim 87 (new): The method of Claim 85, being performed by a program which interfaces with an operating system which itself also performs the method of Claim 86.

Claim 88 (new): The method of Claim 85, wherein objects satisfying one or more of the requirements are flat structures.

Application No. : 09/242,383
Amdt. Dated : January 15, 2004
Reply To Office Action Of : August 15, 2003

Claim 89 (new): The method of Claim 85, wherein objects of the string class for length limited text are stored in particular memory locations at run time which are not part of the heap memory.

Claim 90 (new): The method of Claim 85, wherein the objects are polymorphic.

Claim 91 (new): The method of Claim 90, wherein polymorphism is achieved by providing a data field for each object which identifies its class, with a different action being associated with different data field values.

Claim 92 (new): The method of Claim 91, wherein the data field is a part of the representation of another data item within a machine word.

Claim 93 (new): The method of Claim 92, wherein the same source code is used, irrespective of the character code system and character code width being used, by using aliases for class names that are character code independent.

Claim 94 (new): The method of Claim 93, wherein the source code using text strings is written in a manner independent of the strings' actual ASCII or Unicode implementation by using a system of aliases for class names.

Claim 95 (new): The method of Claim 85, wherein objects have information on the length of the data they contain and hence have no '0' terminator.

Claim 96 (new): Computer software which allows objects of the string class to be manipulated or accessed by a program using an object oriented operating system, wherein the program handles all such objects according to one or more of the following requirements:

- (a) objects of the string class for literal text are handled as belonging to a class in which a pointer points to the memory location where the text string is stored;
- (b) objects of the string class for length limited text are handled as belonging to a class in which a buffer stores text of a predetermined length requiring a limited subset of available memory management functions; and
- (c) objects of the string class using heap memory are handled as belonging to a class requiring the full set of available memory management functions.

Claim 97 (new): The computer software of Claim 96, being an object oriented operating system.

Application No. : 09/242,383
Amdt. Dated : January 15, 2004
Reply To Office Action Of : August 15, 2003

Claim 98 (new): The computer software of Claim 96, being a program which is capable of interfacing with the object oriented operating system of Claim 97.

Claim 99 (new): The computer software of Claim 96, wherein in which objects satisfying one or more of the requirements are flat structures.

Claim 100 (new): The computer software of Claim 96, wherein objects of the string class for length limited text are stored in particular memory locations at run time which are not part of the heap memory.

Claim 101 (new): The computer software of Claim 96, wherein the objects are polymorphic.

Claim 102 (new): The computer software of Claim 101, wherein polymorphism is achieved by providing a data field for each object which identifies its class, with a different action being associated with different data field values.

Claim 103 (new): The computer software of Claim 102, wherein the data field is a part of the representation of another data item within a machine word.

Claim 104 (new): The computer software of Claim 103, wherein the same source code is used, irrespective of the character code system and character code width being used, by using aliases for class names that are character code independent.

Claim 105 (new): The computer software of Claim 104, wherein the source code using text strings is written in a manner independent of the strings' actual ASCII or Unicode implementation by using a system of aliases for class names.

Claim 106 (new): The computer software of Claim 96, wherein objects have information on the length of the data they contain and hence have no '0' terminator.